

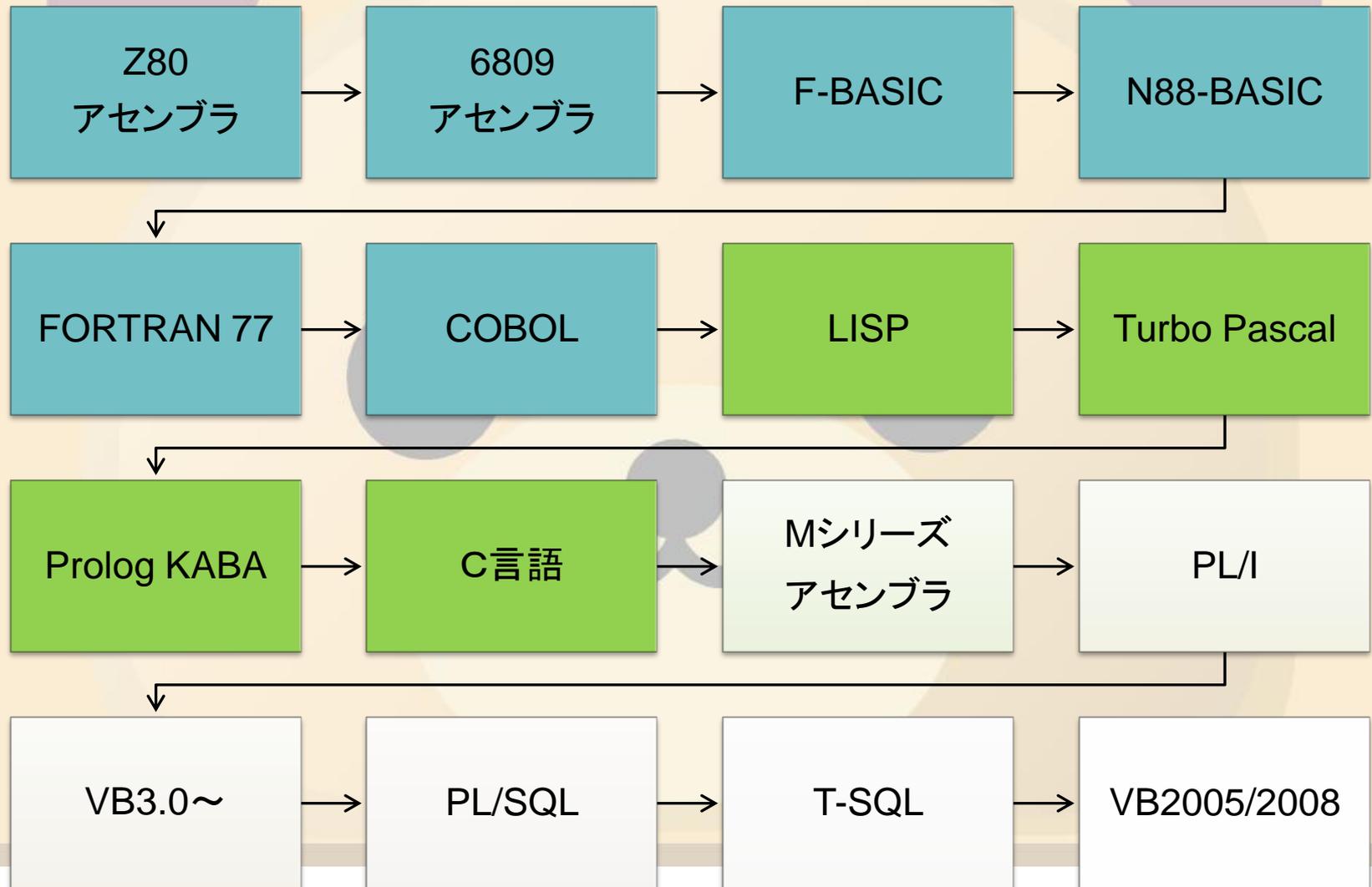
# MVP for VB が語る C# 入門

2008.08.09

初音 玲



# 自己紹介



index

Microsoft Visual C# の第一印象

SQL Serverを使ってみた

Webアプリを作ってみた

Visual Basic と Visual C#



index

Microsoft Visual C# の第一印象

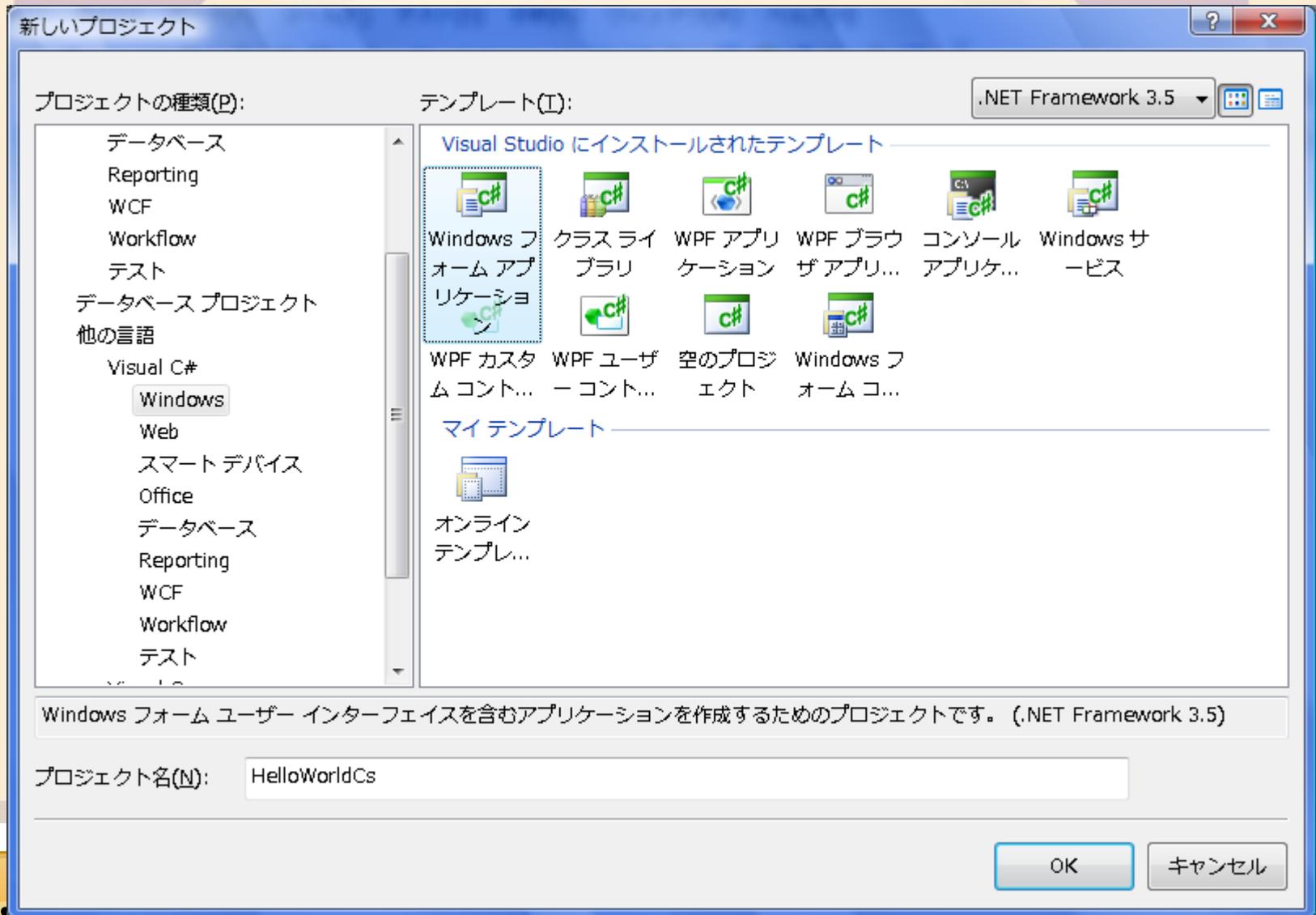
SQL Serverを使ってみた

Webアプリを作ってみた

Visual Basic と Visual C#



# はじめてのC#



# 初期自動生成コード

HelloWorld.cs

HelloWorld.cs [デザイン]

スタートページ

オ

▼ HelloWorldCs.HelloWorld

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

namespace HelloWorldCs
{
    public partial class HelloWorld : Form
    {
        public HelloWorld()
        {
            InitializeComponent();
        }
    }
}
```



# イベントプロシージャ

プロパティ

HelloWorld System.Windows.Forms.Form

ResizeEnd  
RightToLeftChanged  
RightToLeftLayoutCh  
Scroll  
**Shown**  
SizeChanged  
StyleChanged  
SystemColorsChange

Shown  
フォームが最初に表示されたときに発生します。

HelloWorld.Designer.cs

```
this.Shown += new System.EventHandler(this>HelloWorld_Shown);
```

# Hello World!

```
namespace HelloWorldCs
{
    public partial class HelloWorld : Form
    {
        public HelloWorld()
        {
            InitializeComponent();
        }

        private void HelloWorld_Shown(object sender, EventArgs e)
        {
            this.Display_Label.Text = "Hello World!";
        }
    }
}
```



ここに戸惑った！

IDEにフォーマットさせると行数が増える

イベントプロシージャとイベントの関連付け

プロパティの指定や代入文

行の終わりは; (そう言えばCも)

index

Microsoft Visual C# の第一印象

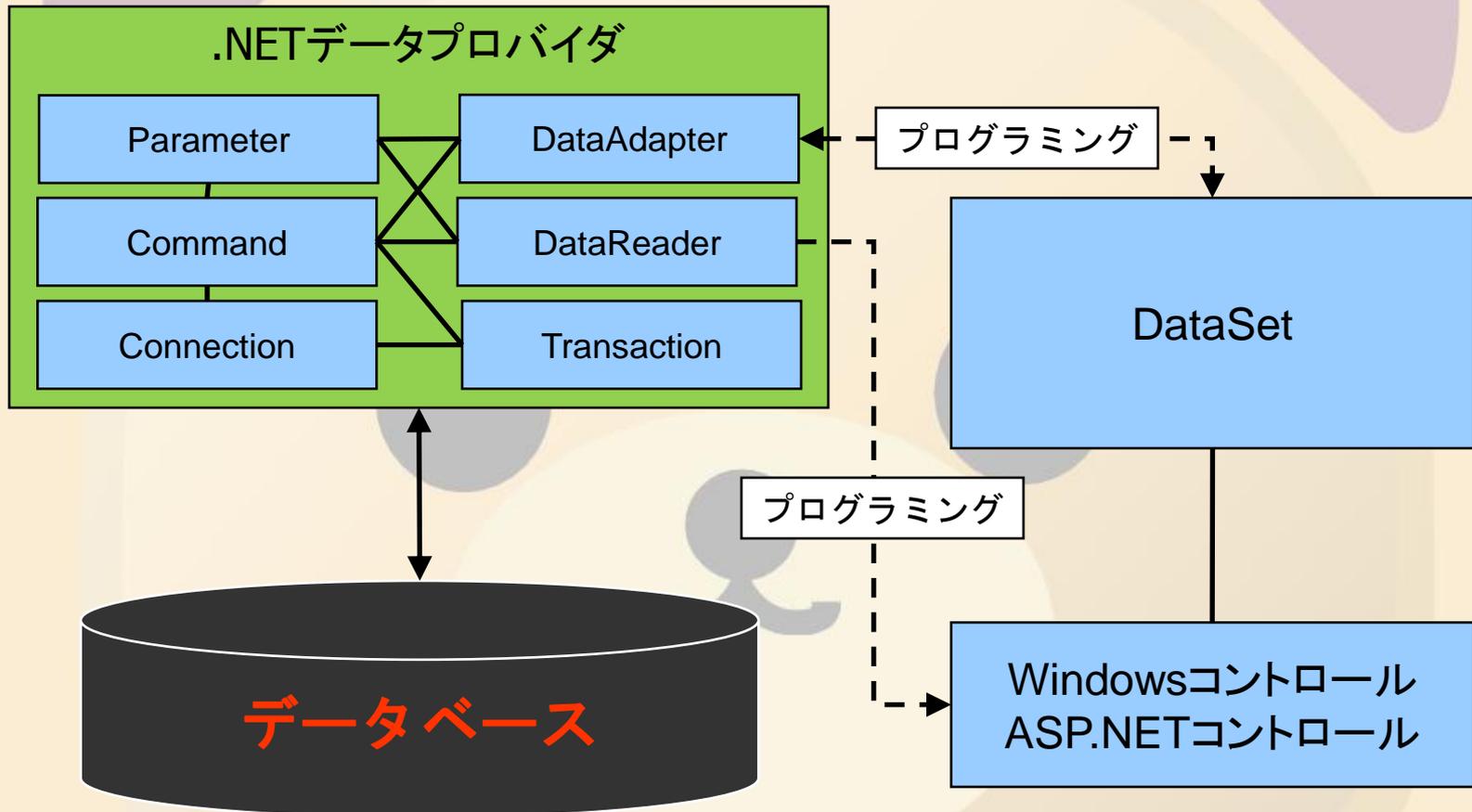
SQL Serverを使ってみた

Webアプリを作ってみた

Visual Basic と Visual C#



# ADO.NETの基本的な構造



# Connection

```
private void Connection_Click(object sender, EventArgs e) {  
    SqlConnection cn = new SqlConnection();  
    this.Cursor = Cursors.WaitCursor;  
    try {  
        cn.ConnectionString = @"User Id=sa;" +  
                               @"Password=;" +  
                               @"Initial Catalog=pubs;" +  
                               @"Data Source=localhost\SQLEXPRESS;";  
        cn.Open();  
    } catch (Exception ex) {  
        MessageBox.Show(ex.Message, this.Text, .....);  
    } finally {  
        this.Cursor = Cursors.Default;  
    }  
}
```



# データソースとDataSetクラスの対応付け

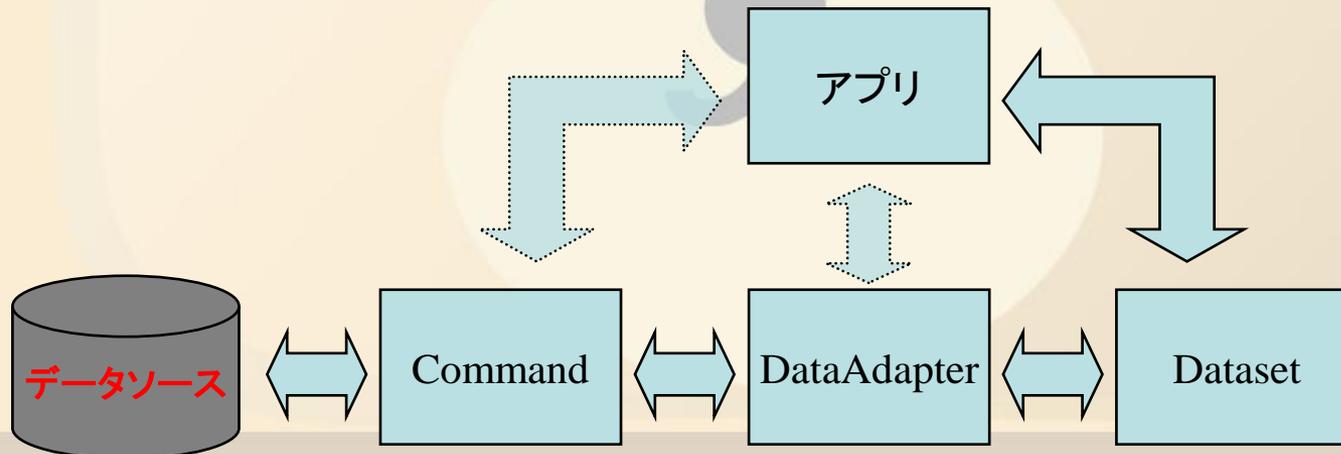
.NETデータプロバイダには、更新可能セットがない

DataSetは仮想的なデータベース

- もちろん更新も可能

.NETデータプロバイダとDataSetの相互乗り入れ

- DataSetによるデータソースの更新可能セットを実現



# DataAdapter

```
using (SqlConnection _cn = new SqlConnection(connectionString)) {
    using (SqlCommand _cmd = new SqlCommand(sqlString, _cn)) {
        _cmd.Transaction = _tr;    #####重要###
        using (SqlDataAdapter _da = new SqlDataAdapter(_cmd)) {
            using (SqlCommandBuilder cb = new SqlCommandBuilder(_da)) {
                _da.UpdateCommand = cb.GetUpdateCommand();
                _da.InsertCommand = cb.GetInsertCommand();
                _da.DeleteCommand = cb.GetDeleteCommand();
                Boolean isOK = False;
                try {
                    _da.Fill(Ds, "employee");
                    isOK = True;
                } catch (Exception ex) {
                    MessageBox.Show(ex.Message, ".....");
                } finally {
                    if (isOK) {
                        _tr.Commit();
                    } else {
                        _tr.Rollback();
                    }
                }
            }
        }
    }
}
```



index

Microsoft Visual C# の第一印象

SQL Serverを使ってみた

Webアプリを作ってみた

Visual Basic と Visual C#



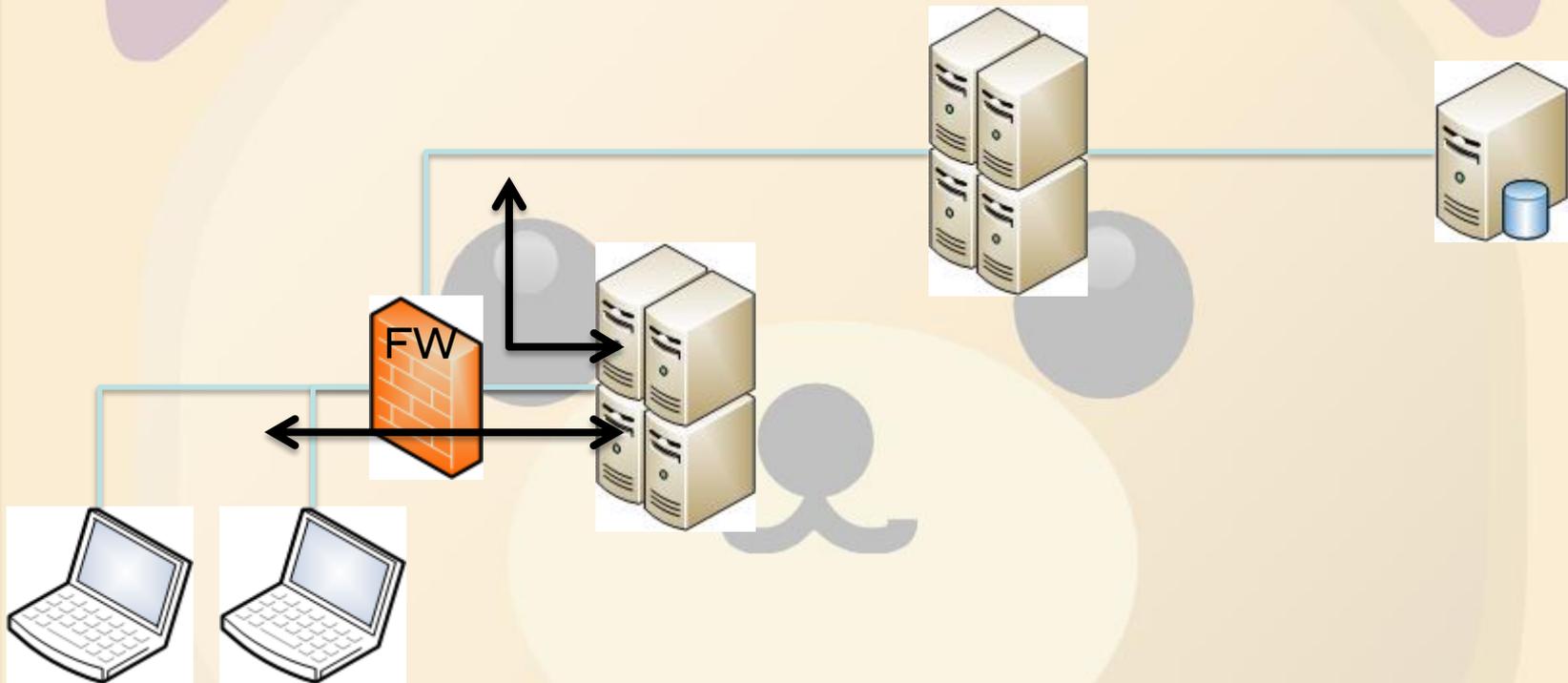
# Webアプリ+Webサービス

ブラウザ

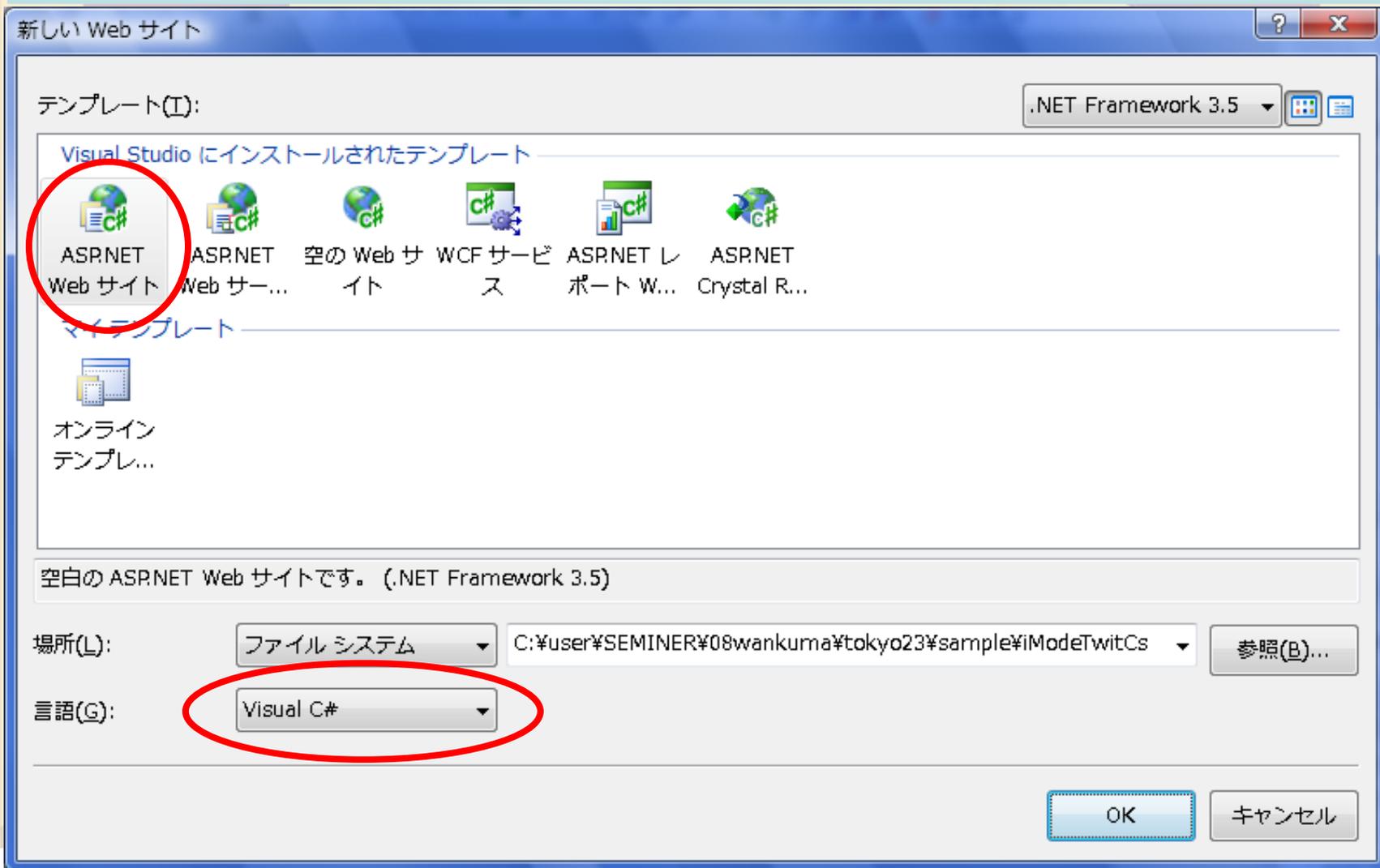
Webアプリ

XML Webサービス

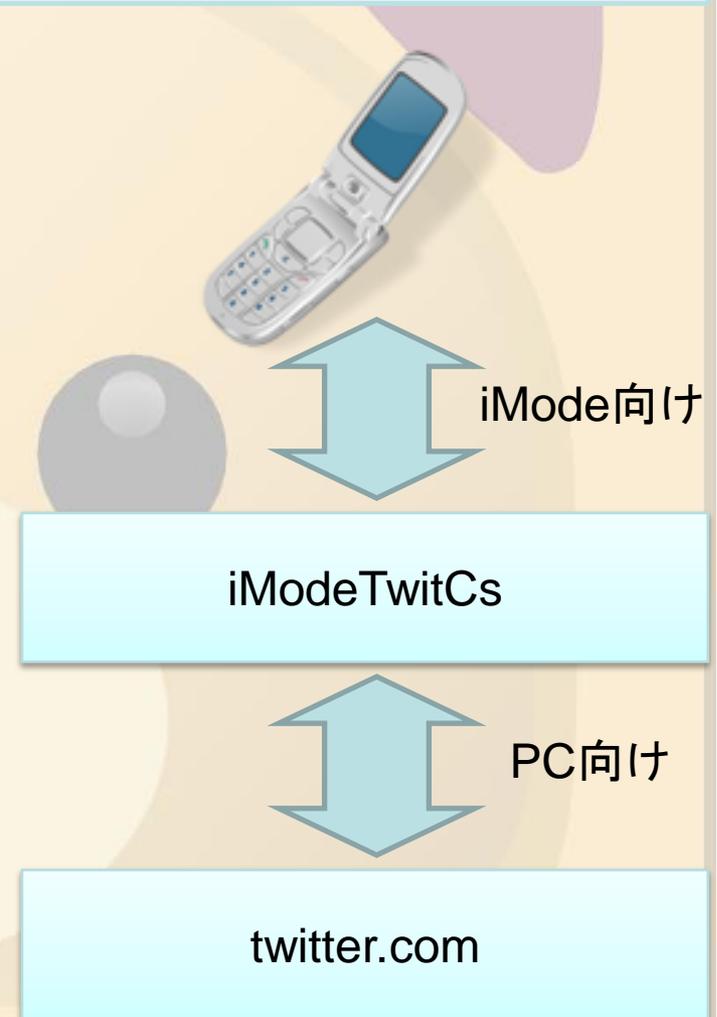
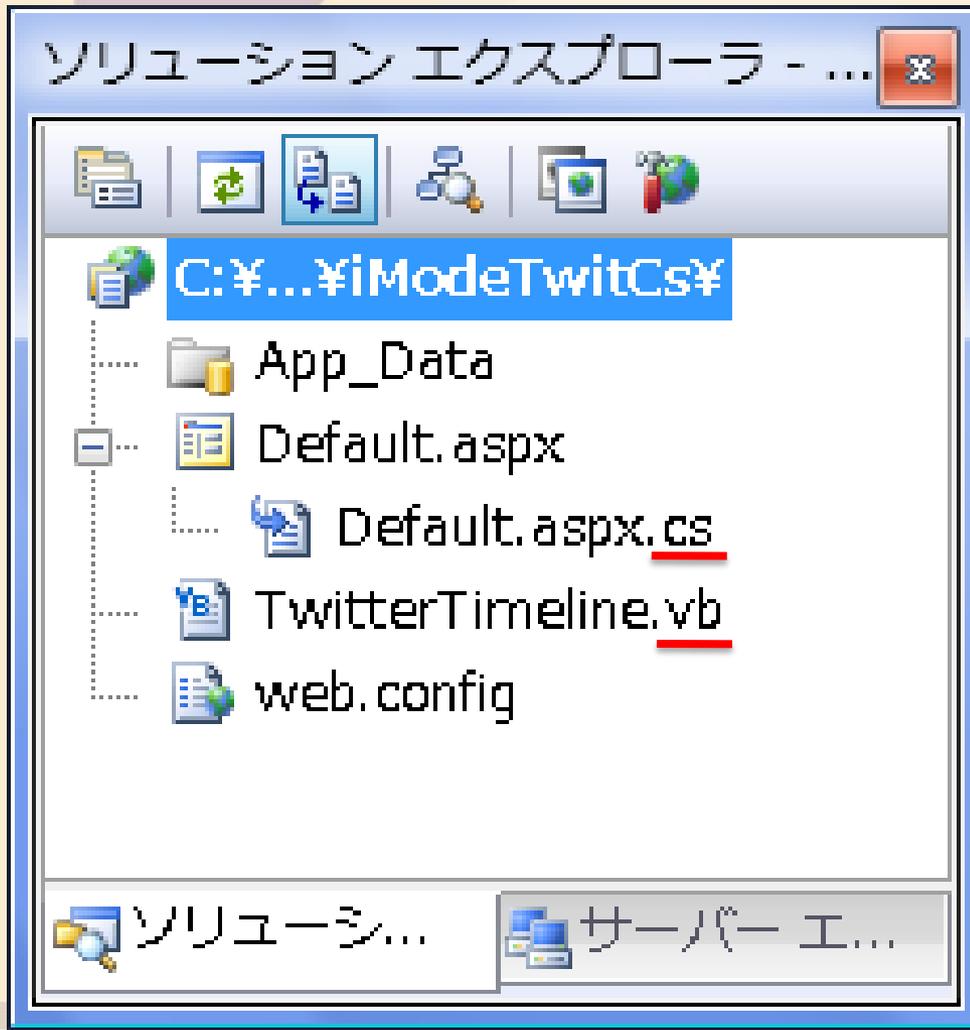
DB



# はじめてのC# - Webアプリ編



# マルチ言語開発



# Webサービスの配置について

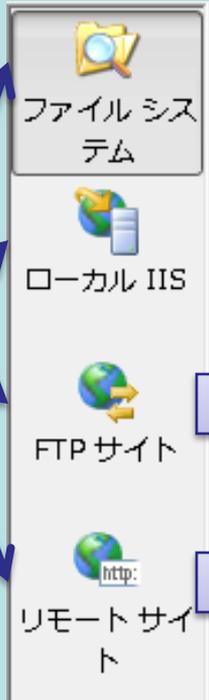
開発環境

実行環境

サイトの発行

MSIL

ソース



MSIL

手動XCOPY

MSIL

初回時

native

MSIL

手動XCOPY

MSIL

初回時

native

FTP

MSIL

初回時

native

HTTP

MSIL

初回時

native

ソース

手動XCOPY

初回時

MSIL

初回時

native



# Webサイトの発行

Webアプリのときは、aspxファイルの内容をサイト上で更新できるかを決定

Web サイトの発行

ターゲットの場所(L): (http://...、https://... またはドライブ:\\*)

!Documents\Visual Studio 2008\Projects\WankumaTokyo22\PrecompiledWeb\WankumaTokyo22

このプリコンパイルされたサイトを更新可能にする(A)

固定名およびシングル ページ アセンブリを使用する(E)

プリコンパイル済みアセンブリで厳密な名前を有効にする(S)

◎ 厳密な名前

キー ファイルの場所:

遅延署名する(D)

◎ キー コンテナを使用する(I)

キー コンテナ:

アセンブリを AllowPartiallyTrustedCallerAttribute (A)

ASPX

dll (aspx.csのMSIL)

JITコンパイル

テンポラリアセンブリ



# プロジェクトと仮想フォルダの関係について

## ソリューション

プロジェクト1

プロジェクト2

プロジェクト3

プロジェクト4

## IIS

仮想フォルダ1

仮想フォルダ2

仮想フォルダ3

仮想フォルダ4

WP

WP

WP

WP

index

Microsoft Visual C# の第一印象

SQL Serverを使ってみた

Webアプリを作ってみた

Visual Basic と Visual C#



# Visual Basic と Visual C#

動作フレームワークは、.NET Framework

- 画面や各機能

IDEは、Visual Studio

- キーアサイン、デバッグなどの機能

行の終わり

- 改行 or 行終端文字





わんくま同盟 東京勉強会 #23 - C# Day